Optimal Correlated Equilibria in General-Sum Extensive-Form Games: Fixed-Parameter Algorithms, Hardness, and Two-Sided Column-Generation

Brian Hu Zhang[1], Gabriele Farina[1], Andrea Celli[2], Tuomas Sandholm[1,3,4,5]

[1] Computer Science Department, Carnegie Mellon University [2] Computing Sciences Department, Bocconi University [3] Strategic Machine, Inc. [4] Strategy Robot, Inc. [5] Optimized Markets, Inc.

{bhzhang, gfarina, sandholm}@cs.cmu.edu; andrea.celli2@unibocconi.it

OUR CONTRIBUTIONS

- We introduce the **correlation DAG**, a new representation and algorithm for finding optimal correlated equilibria in extensive-form games
  - Related to the **team belief DAG** [Zhang, Farina, & Sandholm '22], also a poster at this workshop
  - Immediately yields an **LP-based algorithm** for optimal correlation
  - Recovers **fixed-parameter guarantees** for its size and complexity
- We also introduce a **two-sided column generation** algorithm, that builds upon earlier column generation algorithms [Farina, Celli, Gatti, & Sandholm '21]

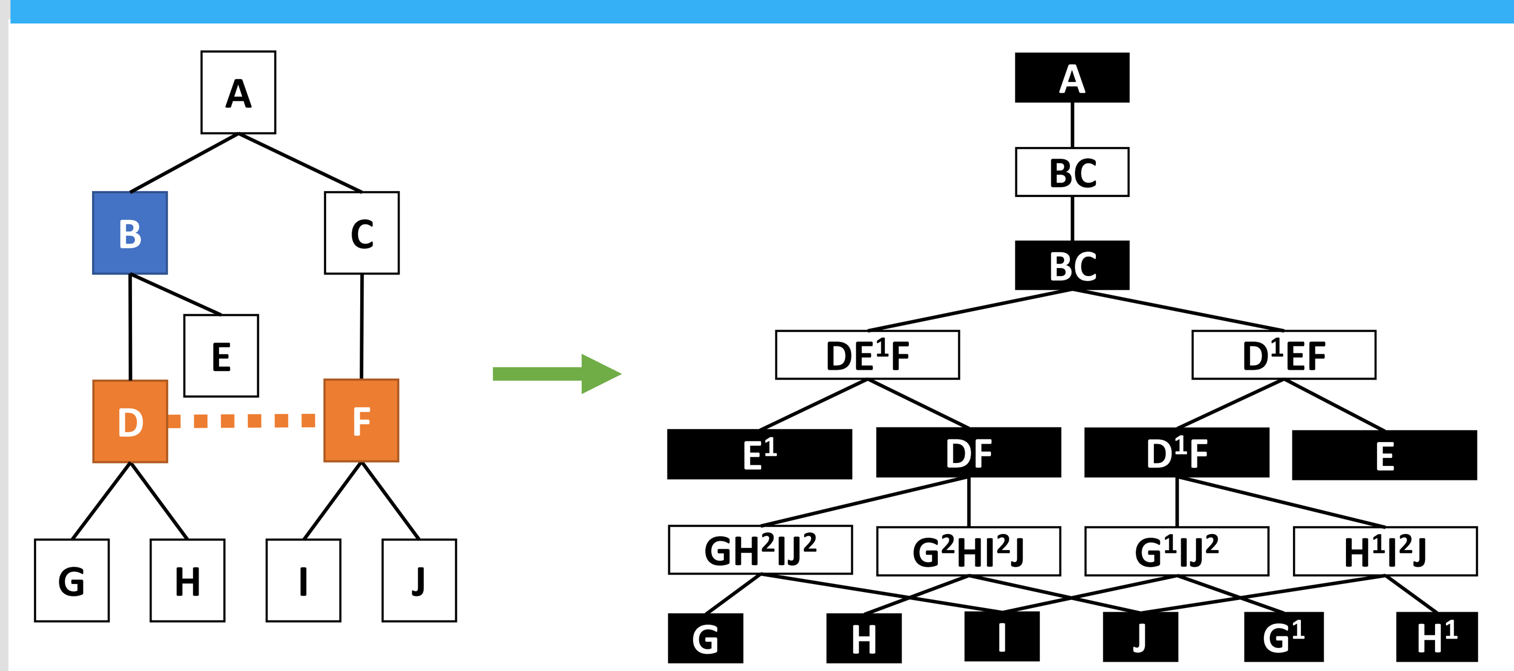
NOTIONS OF CORRELATED EQUILIBRIA IN EFGs

- Best thought of as games played through a **mediator** who has the power to recommend, but not enforce, behavior. Several reasonable notions:

		When do the players decide whether to deviate?	
		Before seeing the recommendation ("coarse correlated")	After seeing the recommendation ("correlated")
How do the players receive recommendations?	Whole strategy at the start of the game ("normal form")	**NFCCE** [Moulin & Vial '78]	**NFCE** [Aumann '74] *Not covered by this paper*
	One recommended move at a time ("extensive form")	**EFCCE** [Farina, Bianchi, & Sandholm '20]	**EFCE** [von Stengel & Forges '08]

- Computing **optimal equilibria** under some given utility function (e.g., social welfare) is **NP-hard** in all four notions, even with two players [von Stengel & Forges '08; Farina, Bianchi, & Sandholm '20]

CORRELATION DAG



- Represents the decision problem faced **by the mediator.** At each timestep, the mediator…
  - selects a **prescription** of recommendations (one action for every possible information set of every player), and then
  - observes the **public information**
- Symbols like **H¹** and **I²** are **trigger histories.** e.g., **H¹** denotes the possibility that everyone was recommended to play to **H** *except* player **1**, who was recommended to play something else

THEORETICAL RESULTS

- Correlation DAG is correct for any number of players in any game
- Size of DAG (and hence LP) is bounded by…
  - $O^*\left((b+1)^k\right)$ for NFCCE, ← Same as team belief DAG!
  - $O^*\left((b+d)^k\right)$ for EFCCE, and
  - $O^*\left((bd)^k\right)$ for EFCE

  where:
  - b is the branching factor (which can be assumed to be 2 WLOG if the game has public actions),
  - d is the depth, and
  - k is the greatest number of *player private states* (summed across all players) per *public state*
- The extra parameter in EFCCE and NFCCE is **fundamental**: under standard complexity assumptions, we show that **there is no algorithm with runtime** $O^*(f(b,k))$ (for any function f) for optimizing over the space of EFCCEs or EFCEs

TWO-SIDED COLUMN GENERATION

- The space of correlated strategies **in a two-player game** can be represented as $\operatorname{co}\{xy^\top : x \in \operatorname{co}(X), y \in \operatorname{co}(Y)\}$ where X, Y are the pure strategy spaces of the two players (easy to represent) and co is the convex hull
- **Idea #1** [Celli & Gatti '18]: use **column generation**: in each iteration t, generate a vertex $x_t y_t^\top$ for $x_t \in X, y_t \in Y$, and reoptimize on $\operatorname{co}\{x_\tau y_\tau^\top : \tau \le t\}$.
- **Idea #2** [Farina, Celli, Gatti, & Sandholm '21]: in each iteration t, generate a new vertex $x_t \in X$, and reoptimize over $\operatorname{co}\{x_\tau y^\top : \tau \le t, y \in \operatorname{co}(Y)\}$.
  - Generates a tighter model than Idea #1 with fewer iterations.
  - Intuitive idea: One player selects a pure strategy, then the other selects a mixed strategy conditional on the first player's pure strategy.
- **Idea #3 [this paper]**: in each iteration t, generate new vertices $x_t \in X, y_t \in Y$, and reoptimize over $\operatorname{co}(\{x_\tau y^\top : \tau \le t, y \in \operatorname{co}(Y)\} \cup \{x y_\tau^\top : \tau \le t, y \in \operatorname{co}(X)\})$.
  - Generates an even tighter model!
  - Intuitive idea: Allow **both** players to take the role of the normalizing player, and allow mixing over these options

COMPARISON OF METHODS

Criterion	Correlation DAG	Column generation
Theoretical runtime bound	**Fixed-parameter tractable**	Exponential
Anytime algorithm	No	**Yes**
Memory usage	Fixed-parameter tractable	**Polynomial** (in number of iterations)
Practical performance when k is small	**Faster**	Slower
Practical performance when k is large	Slower (or out of memory)	**Faster**

EXPERIMENTS

Game			Concept	$\lvert\mathcal{E}^c\rvert$	Value	[vSF08]	Column generation [FCGS21]	Column generation This paper	DAG This paper
^{2}B222	$\lvert\mathcal{Z}\rvert$	1,072	NFCCE	7,093	0.000	0.07s	0.97s	0.24s	0.02s
	$\lvert\Sigma\rvert$	11,049	EFCCE	22,821	−0.525	0.15s	1m 56s	19.57s	0.05s
	k	8	EFCE	20,226	−0.525	0.28s	36m 46s	2m 1s	0.17s
^{2}B323	$\lvert\mathcal{Z}\rvert$	191,916	NFCCE	1,060,277	0.000	1m 5s	> 6h	> 6h	2.82s
	$\lvert\Sigma\rvert$	3,893,341	EFCCE	5,820,762	−0.375	2m 14s	> 6h	> 6h	32.94s
	k	12	EFCE	7,236,102	−0.375	oom	> 6h	oom	1m 55s
^{2}B324	$\lvert\mathcal{Z}\rvert$	969,516	NFCCE	6,531,150	0.000	oom	oom	oom	26.27s
	$\lvert\Sigma\rvert$	26,443,741	EFCCE	43,551,248	−0.489	oom	oom	oom	13m 15s
	k	12	EFCE	49,063,556	−0.489	oom	oom	oom	57m 12s
^{2}S123	$\lvert\mathcal{Z}\rvert$	2,376	NFCCE	19,187	13.636	0.26s	18.18s	3.03s	0.04s
	$\lvert\Sigma\rvert$	33,633	EFCCE	75,479	10.000	0.59s	1h 1m	5m 52s	0.23s
	k	12	EFCE	52,559	10.000	1.22s	1h 11m	7m 6s	0.65s
^{2}S133	$\lvert\mathcal{Z}\rvert$	5,632	NFCCE	46,755	18.182	1.05s	3m 21s	9.01s	0.04s
	$\lvert\Sigma\rvert$	95,768	EFCCE	179,571	15.000	1.94s	> 6h	1h 26m	1.51s
	k	12	EFCE	165,859	15.000	6.45s	> 6h	> 6h	2.46s
^{2}RS12	$\lvert\mathcal{Z}\rvert$	400	NFCCE	10,366	6.010	n/a	0.04s	0.04s	0.02s
	$\lvert\Sigma\rvert$	613	EFCCE	10,366	6.010	n/a	0.08s	0.06s	0.01s
	k	15	EFCE	8,846	6.010	n/a	0.54s	0.13s	0.01s
^{2}RS22	$\lvert\mathcal{Z}\rvert$	484	NFCCE	34,947	7.188	n/a	0.08s	0.28s	0.20s
	$\lvert\Sigma\rvert$	701	EFCCE	34,947	7.176	n/a	0.13s	0.08s	0.20s
	k	15	EFCE	31,503	7.176	n/a	0.56s	0.41s	0.16s
^{2}RS23	$\lvert\mathcal{Z}\rvert$	4,096	NFCCE	oom	10.961	n/a	2.63s	3.12s	oom
	$\lvert\Sigma\rvert$	13,277	EFCCE	oom	10.820	n/a	1m 51s	56.31s	oom
	k	44	EFCE	oom	10.791	n/a	> 6h	6m 35s	oom

We tested our algorithms against two earlier algorithms:
- [FCGS21] is Farina, Celli, Gatti, & Sandholm ['21] ("Idea #2")
- [vSF08] is the LP relaxation of von Stengel & Forges ['08], which only works in **triangle-free games** [Farina & Sandholm '20]

Game instances tested:
- *Bhwr* is 2-player Battleship [Farina, Ling, Fang, & Sandholm '19] on a grid of size $h \times w$, one unit-size ship per player, and r rounds
- *Snbr* is a simplified version of the 2-player Sherrif of Nottingham [Farina, Ling, Fang, & Sandholm '19] game, with n items for the smuggler, a maximum bribe of b, and r rounds of bargaining
- 2RS*iT* is a ride-sharing game. It is played on finite graph. Two drivers seek to earn points by reaching specific nodes of the graph and serving the requests at those nodes. Parameter i specifies the graph configuration, while T is the time horizon

FURTHER RESEARCH

- Can we get a "best-of-both-worlds" algorithm that has the worst-case performance of our algorithm but the convenient properties (e.g., anytime-ness) of column generation, and is superior to both in practice?
- Can this technique result in efficient algorithms in *triangle-free games* [Farina & Sandholm '20] as well?
- Can regret minimization be used for **optimal** correlation through the correlation DAG?

Acknowledgements: This material is based on work supported by the National Science Foundation under grants IIS-1718457, IIS-1901403, IIS-1943607, and CCF-1733556, and the ARO under award W911NF2010081.